CLAIMS

1. A method for processing a signal value in a digital signal processor, the method comprising the step of:

5   in response to a single instruction that specifies at least a signal value and a despreading code, multiplying the signal value by the despreading code.

2. A method as defined in claim 1 further comprising the step of adding

10 the result of the multiplying to a result from a previous multiplication.

3. A method as defined in claim 1, wherein the despreading code has a spreading factor divisible by 4.

15 4. A method as defined in claim 1, wherein the despreading code is divided into code segments, each code segment having comprising a 2 bit complex code comprising 1 real bit and 1 imaginary bit.

5. A method as defined in claim 4, wherein a set code bit represents a

20 value of −1 and a clear code bit represents a value of +1.

6. A method as defined in claim 1, wherein the signal value comprises 16 bits.

25 7. A method as defined in claim 6, wherein the signal value comprises 8 real bits and 8 imaginary bits.

537698.2

8.   A method for calculating a data set in a digital signal processor, the method comprising the steps of:

in response to one or more instructions that specify at least a signal

5   value and a set of codes:

for each one of the set of codes multiplying the signal value by one of the set of codes;

summing results of the multiplying; and

producing a data set resulting from the summing.

10

9.   A method as defined in claim 8, wherein the summing comprises summing results of the multiplying with the results of a multiplying by a previous set of codes.

15   10.   A method as defined in claim 8, wherein the set of codes has a spreading factor divisible by 4.

11.   A method as defined in claim 8, wherein each one of the set of codes is a 2 bit complex code comprising 1 real bit and 1 imaginary bit.

20

12.   A method as defined in claim 11, wherein a set code bit represents a value of −1 and a clear code bit represents a value of +1.

13.   A method as defined in claim 8, wherein the signal value comprises

25   16 bits.

14.   A method as defined in claim 13, wherein the signal value comprises 8 real bits and 8 imaginary bits.

15.   A digital signal processor comprising:

5   a memory for storing instructions and operands for digital signal computations;

          a program sequencer for generating instruction addresses for fetching selected ones of said instructions from said memory; and

          a computation block comprising a register file for temporary storage

10   of operands and results and an execution block for executing a decoding instruction that specifies a data signal and a code, said execution block comprising a complex multiply and accumulate engine for multiplying portions of the data signal by the code and accumulating the results.

15   16.   A digital signal processor as defined in claim 15, wherein, in response to an execution of the decoding instruction the digital signal processor:

          performs a set of complex multiplies on portions of the data signal and portions of the code; and

          sums the results of the complex multiplies.

20

17.   A method as defined in claim 15, wherein the code has a spreading factor divisible by 4.

18.   A method as defined in claim 15, wherein the code is divided into

25   code segments, each code segment having comprising a 2 bit complex code comprising 1 real bit and 1 imaginary bit.

19.   A method as defined in claim 18, wherein a set code bit represents a value of −1 and a clear code bit represents a value of +1.

20.   A method as defined in claim 15, wherein the data signal comprises 16 bits.

21.   A method as defined in claim 20, wherein the data signal comprises 8 real bits and 8 imaginary bits.

22.   A method for calculating output data in a digital signal processor, the method comprising the steps of:

in response to one or more instructions that specify at least a set of complex first operands each one of the first operands comprising 8 real bits and 8 imaginary bits and a set of complex second operands each one of the second operands comprising 1 real bit and 1 imaginary bit:

for each one of the second operands performing a complex multiplication of one of the first operands by one of the second operands;

summing results of the multiplying over the set of second operands; and

producing as an output a set of data resulting from the summing.

23.   A method as defined in claim 22, wherein a set bit in one of the second operands represents a value of −1 and a clear bit in one of the second operands represents a value of +1.

24.   A method as defined in claim 22, wherein the set of complex second operands comprises a despreading code.

25.   A method as defined in claim 22, wherein the set of complex first

5   operands comprises an incoming data signal.

26.   A method as defined in claim 25, wherein the incoming data signal is a voice transmission signal.

10   27.   A method for processing a signal value in a digital signal processor, comprising the step of:

in response to a complex signal value and a two bit complex code segment specified by an instruction, performing a complex multiply of the signal value by the code segment to provide a processed data value.

15

28.   A method for processing signal values in a digital signal processor comprising the steps of:

(a)   in response to a set of complex signal values and a corresponding set of complex code segments specified by an instruction,

20   performing a complex multiply of each signal value by a corresponding code segment to provide a set of intermediate values; and

(b)   performing complex addition of the intermediate values to provide a processed signal value.

25   29.   A method as defined in claim 28 further comprising the steps of repeating steps (a) and (b) for a plurality of sets of complex signal values to provide a stream of processed signal values.

537698.2

30.   A method as defined in claim 28 further wherein each of the complex code segments is a two bit complex code.

5

10

537698.2